IN THE CLAIMS

Please amend claims 31-33, 38-40, and 45. Claims 1-30 were previously canceled.

All pending claims and their present status are produced below.

1-30. (Canceled)

31. (Currently Amended) A method for calculating application verb response times, comprising:

receiving a data packet containing ~~application-layer data related to a network application~~ data of an application protocol;

identifying ~~a portion of~~ an instance of an application verb in the data packet, the application verb being a ~~specific application-layer~~ transaction type within the application protocol;

updating a state machine based on the ~~portion of the~~ instance of the application verb, the state machine comprising a current state of the application;

determining whether the ~~portion~~ instance of the application verb represents a completed response ~~of the application verb~~ based on the updated state machine, a completed response being a response to a previous instance of the application verb with no further response to the previous instance of the application verb being expected; and

responsive to determining that the ~~portion~~ instance of the application verb represents a completed response ~~of the application verb~~, calculating a response time associated with the application verb between ~~a request~~ the previous instance of the application verb and the ~~completed response~~ instance of the application verb.

32. (Currently Amended) The method of claim 31, wherein determining whether the ~~portion~~ instance of the application verb represents a completed response ~~of the application verb~~ based on the updated state machine further comprises checking for retransmissions, out-of-sequence packets, errors, and complications.

33. (Currently Amended) The method of claim 31, further comprising identifying a protocol identifier associated with the application protocol and determining a number of known application verbs associated with the protocol identifier.

34. (Previously Presented) The method of claim 33, further comprising allocating memory for a data structure based on the number of known application verbs associated with the protocol identifier.

35. (Previously Presented) The method of claim 31, wherein a current node of the state machine and a last seen application verb are stored as a bit vector.

36. (Previously Presented) The method of claim 31, wherein the calculated response time associated with the application verb is mapped to a RMON tree.

37. (Previously Presented) The method of claim 31, wherein calculating a response time associated with the application verb is performed in real-time.

38. (Currently Amended) A computer-readable medium having a computer program product for calculating application verb response times, comprising:

computer code for receiving a data packet containing ~~application-layer data related to a network application~~ data of an application protocol;

computer code for identifying ~~a portion of~~ an instance of an application verb in the data packet, the application verb being a ~~specific application-layer~~ transaction type within the application protocol;

computer code for updating a state machine based on the ~~portion of the~~ instance of the application verb, the state machine comprising a current state of the application;

computer code for determining whether the ~~portion~~ instance of the application verb represents a completed response ~~of the application verb~~ based on the updated state machine, a completed response being a response to a previous instance of the application verb with no further response to the previous instance of the application verb being expected; and

computer code for, responsive to determining that the ~~portion~~ instance of the application verb represents a completed response ~~of the application verb~~, calculating a

response time associated with the application verb between ~~a request~~ the previous instance of the application verb and the ~~completed response~~ instance of the application verb.

39. (Currently Amended) The computer-readable medium of claim 38, wherein computer code for determining whether the ~~portion~~ instance of the application verb represents a completed response ~~of the application verb~~ based on the updated state machine further comprises computer code for checking for retransmissions, out-of-sequence packets, errors, and complications.

40. (Currently Amended) The computer-readable medium of claim 38, wherein the computer program product further comprises computer code for identifying a protocol identifier associated with the application protocol and computer code for determining a number of known application verbs associated with the protocol identifier.

41. (Previously Presented) The computer-readable medium of claim 40, wherein the computer program product further comprises computer code for allocating memory for a data structure based on the number of known application verbs associated with the protocol identifier.

42. (Previously Presented) The computer-readable medium of claim 38, wherein a current node of the state machine and a last seen application verb are stored as a bit vector.

43. (Previously Presented) The computer-readable medium of claim 38, wherein the calculated response time associated with the application verb is mapped to a RMON tree.

44. (Previously Presented) The computer-readable medium of claim 38, wherein calculating a response time associated with the application verb is performed in real-time.

45. (Currently Amended) A system for calculating application verb response times, comprising:

means for receiving a data packet containing ~~application-layer data related to a network application~~ data of an application protocol;

means for identifying ~~a portion of~~ an instance of an application verb in the data packet, the application verb being a ~~specific application-layer~~ transaction type within the application protocol;

means for updating a state machine based on the ~~portion of the~~ instance of the application verb, the state machine comprising a current state of the application;

means for determining whether the ~~portion~~ instance of the application verb represents a completed response ~~of the application verb~~ based on the updated state machine, a completed response being a response to a previous instance of the application verb with no further response to the previous instance of the application verb being expected; and

means for, responsive to determining that the ~~portion~~ instance of the application verb represents a completed response ~~of the application verb~~, calculating a response time associated with the application verb between ~~a request~~ the previous instance of the application verb and the ~~completed response~~ instance of the application verb.